

# Robust Simultaneous Localization of Nodes and Targets in Sensor Networks Using Range-Only Measurements

Pınar Oğuz-Ekim, *Student Member, IEEE*, João Gomes, *Member, IEEE*,  
João Xavier, *Member, IEEE*, and Paulo Oliveira, *Member, IEEE*,

## Abstract

Simultaneous localization and tracking (SLAT) in sensor networks aims to determine the positions of sensor nodes and a moving target in a network, given incomplete and inaccurate range measurements between the target and each of the sensors. One of the established methods for achieving this is to iteratively maximize a likelihood function (ML), which requires initialization with an approximate solution to avoid convergence towards local extrema. This paper develops methods for handling both Gaussian and Laplacian noise, the latter modeling the presence of outliers in some practical ranging systems that adversely affect the performance of localization algorithms designed for Gaussian noise. A modified Euclidean Distance Matrix (EDM) completion problem is solved for a block of target range measurements to approximately set up initial sensor/target positions, and the likelihood function is then iteratively refined through Majorization-Minimization (MM). To avoid the computational burden of repeatedly solving increasingly large EDM problems in time-recursive operation an incremental scheme is exploited whereby a new target/node position is estimated from previously available node/target locations to set up the iterative ML initial point for the full spatial configuration. The above methods are first derived under Gaussian noise assumptions, and modifications for Laplacian noise are then considered. Analytically, the main challenges to be overcome in the Laplacian case stem from the non-differentiability of  $\ell_1$  norms that arise in the various cost functions. Simulation results confirm that the proposed algorithms significantly outperform existing methods for SLAT in the presence of outliers, while offering comparable performance for Gaussian noise.

The authors are with the Institute for Systems and Robotics – Instituto Superior Técnico (ISR/IST), Lisbon, Portugal. e-mail: poguz@isr.ist.utl.pt.

This research was partially supported by Fundação para a Ciência e a Tecnologia (FCT) through ISR/IST plurianual funding with PIDDAC program funds, project PTDC/EEA-CRO/104243/2008, and grant SFRH/BD/44771/2008.

## I. INTRODUCTION

This work addresses the problem of tracking a single target from distance-like measurements taken by nodes in a sensor network whose positions are not precisely known. The goal is to estimate the positions of all sensors and the target, given only partial or no *a priori* information on the spatial configuration of the network. The ability to track a target is a key component in several scenarios of wireless sensor networks, hence methods that avoid the need for careful calibration of sensor positions are practically relevant.

In [1], [2] SLAT is formulated in a Bayesian framework that resembles the related and well-studied problem of Simultaneous Localization and Mapping (SLAM) in robotics. The *a posteriori* probability density function of sensor/target positions and calibration parameters is recursively propagated in time as more target sightings become available. In [1] these observations are true range measurements obtained through a combination of transmitted acoustic and radio pulses. Alternatives to range include pseudorange and bearing information estimated from camera images [2], or the (somewhat unreliable) Received Signal Strength (RSS) of radio transmissions [3]. In [4] the SLAT problem is also formulated in a Bayesian framework as a general state evolution model under a binary proximity model and solved in a decentralized way using binary sensor networks. Other SLAT-like approaches include localization (calibration) as presented in [5], where positions and orientations of unknown sources and sensors are centrally obtained via ML based on time-of-arrival and angle-of-arrival measurements. Several of the above references emphasize decentralized processing, in line with the local observation model.

When target dynamics are not accounted for, the SLAT problem may be thought of as a special case of Sensor Network Localization (SNL) with a limited set of intersensor measurements. In fact, EDM and related matrix completion methods based on Semidefinite Programming (SDP) have been adopted previously for static SNL (see [6] and references therein). EDM completion for SLAT has been discussed in [3], although the authors ended up pursuing an alternative approximate completion approach based on a variant of Multidimensional Scaling (MDS). Underwater and underground scenarios with uncertainty in anchor positions are considered in [7], and edge-based SDP is proposed to reduce the computational complexity of SNL. In [8] static SNL is formulated as a problem of ML phase retrieval.

In addition to centralized SNL approaches such as [6]–[8], enumerated above, a wealth of results is available on distributed approaches for scenarios where the existence of a central node is inconvenient, e.g., due to congested communications in its vicinity, or excessive vulnerability of the whole infrastructure to failure of that single node [3], [9]–[13]. A two-step approach based on second-order cone programming

relaxation with inaccurate anchor positions is introduced in [9]. In [11] a weighted least-squares algorithm with successive refinement provides both position estimates and their covariances in partially connected scenarios. A distributed weighted MDS method with majorization approximations is applied in [12]. The cost function and majorization technique are similar to the ones used in this paper for ML iterative refinement under Gaussian noise, but initialization relies on prior estimates of sensor positions. A wholly different iterative approach for distributed SNL using barycentric coordinates and Cayley-Menger determinants is developed in [13]. For  $m$ -dimensional Euclidean space a node reinterpolates its position based on estimates from a set of  $m + 1$  neighbors such that it lies in their convex hull.

This paper focuses on centralized SLAT based on plain ML estimation. A basic iterative optimization approach using Majorization Minimization (MM) is first developed for batch estimation, i.e., when all measurements are processed simultaneously. A time recursive method is then obtained by estimating each new target or sensor position as the corresponding range measurements become available, and then iteratively re-optimizing the expanded ML cost function. This continuation scheme only requires initialization of all target/sensor positions at the first time step, which is computationally less complex than doing so for each new sensed target or sensor position. EDM completion for batch estimation is proposed to initialize the iterative ML algorithm with little *a priori* knowledge of sensor/target positions, thus alleviating the problem of convergence to undesirable local extrema.

This approach was proposed in [14] for Gaussian noise, using cost functions for batch and time recursive initialization schemes that match *squared* observations with *squared* estimated ranges. These discrepancies are eliminated in the present paper, such that both initialization and ML refinement operate with cost functions that match plain (non-squared) ranges, leading to improved robustness under strong measurement noise [15]. We use the Source Localization in Complex Plane (SLCP) approach proposed in [16] to obtain a cost function for initialization of incremental target/sensor position estimates that admits an accurate convex relaxation as an SDP.

This work also develops modified versions of initialization and ML refinement for Laplacian noise, which models the presence of outliers in some practical ranging systems that adversely affect the performance of localization algorithms designed for Gaussian noise [1], [17]. This is accomplished by replacing  $\ell_2$  norms with  $\ell_1$  norms in the optimization problems for initialization and refinement that were previously formulated for Gaussian noise, and then performing suitable manipulations to write these in a form that is amenable to general-purpose solvers. A source localization algorithm with  $\ell_1$  norms (SL $\ell_1$ ) is also derived for incremental initialization of the SLAT scheme.

In [17]  $\ell_1$  norms are also used to handle outliers, but the proposed method is very different from the

one developed here, as it relies on linear programming to identify the outliers, and then remove them from consideration when computing the source location. In our work all measurements are kept, as the modulus of range differences that appears in cost functions ensures that outlier terms do not overwhelm the remaining ones as long as the proportion of outliers remains small. Another approach for handling outliers is presented in [18], where the Huber cost function interpolates between  $\ell_1$  and  $\ell_2$  norms. This function is minimized via iterative majorization techniques with *a priori* information on sensor positions, where in each step a majorization subproblem is solved using Costa's algorithm.

The paper is organized as follows. In section II, the SLAT problem is introduced. Section III presents estimation methods for range measurements corrupted by Gaussian noise, namely, EDM initialization, iterative likelihood refinement by MM, and time-recursive updating through incremental estimation of target/sensor positions. Section IV develops similar methods for Laplacian noise. Section V provides simulation results of batch and time recursive approaches. In addition, the performances of initialization techniques are compared with and without outliers. Finally, section VI summarizes the main conclusions.

Throughout, both scalars and individual position vectors in (2D) space will be represented by lowercase letters. Matrices and vectors of concatenated coordinates will be denoted by boldface uppercase and lowercase letters, respectively. The superscript  $T$  ( $H$ ) stands for the transpose (Hermitian) of the given real (complex) vector or matrix. Below,  $\mathbf{I}_m$  is the  $m \times m$  identity matrix and  $\mathbf{1}_m$  is the vector of  $m$  ones. For symmetric matrix  $\mathbf{X}$ ,  $\mathbf{X} \succeq 0$  means that  $\mathbf{X}$  is positive semidefinite.

## II. PROBLEM FORMULATION

The network comprises sensors at unknown positions  $\{x_1, x_2, \dots, x_n\} \in \mathbb{R}^2$ , a set of reference sensors (anchors) at known positions  $\{a_1, a_2, \dots, a_l\} \in \mathbb{R}^2$ , and unknown target positions  $\{e_1, e_2, \dots, e_m\} \in \mathbb{R}^2$ . A central processing node has access to range measurements between target positions and all sensors/anchors, namely,

$$d_{ij} = \|x_i - e_j\| + w_{ij}, \quad d_{kj} = \|a_k - e_j\| + w_{kj},$$

where  $w_{ij}$  and  $w_{kj}$  denote noise terms. A practical system that provides such range measurements is used, e.g., in [1].

*a) SLAT Under Gaussian Noise:* If disturbances are Gaussian, independent and identically distributed (i.i.d.), maximizing the likelihood for the full batch of observations is equivalent to minimizing the cost function

$$\Omega_G(\mathbf{x}) = \sum_{i,j} (\|x_i - e_j\| - d_{ij})^2 + \sum_{k,j} (\|a_k - e_j\| - d_{kj})^2. \quad (1)$$

The full set of unknown sensor and target positions is concatenated into column vector  $\mathbf{x} \in \mathbb{R}^{2(n+m)}$ , the argument of  $\Omega_G$ . The goal of our SLAT approach is to find the set of coordinates in  $\mathbf{x}$  which minimizes (1).

*b) SLAT Under Laplacian Noise:* When the disturbances are Laplacian and i.i.d., thus heavier tailed than Gaussian, maximizing the likelihood amounts to minimizing the cost function

$$\Omega_L(\mathbf{x}) = \sum_{i,j} |||x_i - e_j|| - d_{ij}| + \sum_{k,j} |||a_k - e_j|| - d_{kj}|. \quad (2)$$

When compared with (1), the absence of squares in the summation terms of (2) renders the function less sensitive to outlier measurements  $d_{ij}$  with large deviations from the true ranges.

Due to the nature of this problem the functions  $\Omega_G$  and  $\Omega_L$  are invariant to global rotation, translation and reflection in the absence of anchors. In order to remove ambiguities in the solutions, at least  $l = 3$  non collinear anchors must be considered. As in many other ML problems, the functions  $\Omega_G$  and  $\Omega_L$  are in general nonconvex and multimodal, hence their (approximate) minimization proceeds in two phases, initialization and refinement. The former provides a suitable initial point through EDM completion, which tends to avoid convergence towards undesirable local minimizers of the ensuing iterative refinement algorithms based on MM or weighted-MM.

### III. SLAT UNDER GAUSSIAN NOISE

This section develops algorithms for EDM initialization, MM refinement, and recursive estimation in SLAT under the assumption that measurement noise is i.i.d. and Gaussian. Modifications for i.i.d. Laplacian noise are considered in Section IV. A basic formulation of EDM completion with squared distances is provided first, and will form the basis for the initialization methods described in Sections III-B and IV-A.

#### A. EDM with Squared Distances

The basic EDM completion problem, described below, operates on squared ranges [19], [20]. Even though it is not matched to the likelihood function (1), it will be useful for benchmarking in Section V, as its performance is representative of other popular SNL methods [6], [21] and the SLAT approach of [14].

A partial pre-distance matrix  $\mathbf{D}$  is a matrix with zero diagonal entries and with certain elements fixed to given nonnegative values which are the squared observed distances,  $D_{ij} = d_{ij}^2$ . The remaining elements are considered free. The nearest EDM problem is to find an EDM  $\mathbf{E}$  that is nearest in the

least-squares sense to matrix  $\mathbf{D}$ , when the free variables are not considered and the elements of  $\mathbf{E}$  satisfy  $E_{ij} = \|y_i - y_j\|^2$  for a set of points  $y_i$ . The geometry and properties of EDM (a convex cone) have been extensively studied in the literature [19], [20]. The nearest EDM problem in 2D space is formulated as

$$\begin{aligned}
& \underset{\mathbf{E}}{\text{minimize}} && \sum_{i,j \in \mathcal{O}} (E_{ij} - d_{ij}^2)^2 \\
& \text{subject to} && \mathbf{E} \in \mathcal{E} \\
& && \mathbf{E}(\mathcal{A}) = \mathbf{A} \\
& && \text{rank}(\mathbf{J}\mathbf{E}\mathbf{J}) = 2,
\end{aligned} \tag{3}$$

where

$$\mathbf{J} = (\mathbf{I}_\rho - \frac{1}{\rho} \mathbf{1}_\rho \mathbf{1}_\rho^T), \quad \rho = m + n + l,$$

is a centering operator which subtracts the mean of a vector from each of its components. In (3),  $\mathcal{O}$  is the index set for which range measurements are available. The constraint  $\mathbf{E}(\mathcal{A}) = \mathbf{A}$ , where  $\mathcal{A}$  is the index set of anchor/anchor distances and  $\mathbf{A}_{ij} = \|a_i - a_j\|^2$  is the corresponding EDM submatrix, enforces the known *a priori* spatial information. Matrix  $\mathbf{E}$  belongs to the EDM cone  $\mathcal{E}$  if it satisfies the properties

$$E_{ii} = 0, \quad E_{ij} \geq 0, \quad -\mathbf{J}\mathbf{E}\mathbf{J} \succeq 0. \tag{4}$$

The rank constraint in (3) ensures that the solution is compatible with a constellation of sensor/anchor/target points in  $\mathbb{R}^2$ . Extraction of the set  $y_i$  from  $\mathbf{E}$  is described below. Problem (3) is also known as the penalty function approximation [19] due to the form of the cost function  $\varphi_1(\mathbf{E}) = \sum_{i,j} (E_{ij} - d_{ij}^2)^2$ . Expressing (3) in terms of full matrices and dropping the rank constraint, a compact relaxed SDP formulation is obtained as

$$\begin{aligned}
& \underset{\mathbf{E}}{\text{minimize}} && \|\mathbf{W} \odot (\mathbf{E} - \mathbf{D})\|_F^2 \\
& \text{subject to} && \mathbf{E} \in \mathcal{E}, \mathbf{E}(\mathcal{A}) = \mathbf{A},
\end{aligned} \tag{5}$$

where  $\mathbf{W}$  is a mask matrix with zeros in the entries corresponding to free elements of  $D_{ij} = d_{ij}^2$ , and ones elsewhere. Combined with the Hadamard product  $\odot$ , and Frobenius norm  $\|\cdot\|_F$ , this replaces the summation in (3) over the observed index set  $\mathcal{O}$ . From here on, we will call this method EDM with squared ranges (EDM-SR).

### B. SLAT Initialization: EDM with Plain Distances

Instead of trying to match squared distances, we can apply EDM completion to plain distances as

$$\begin{aligned}
& \underset{\mathbf{E}}{\text{minimize}} && \sum_{i,j} (\sqrt{E_{ij}} - d_{ij})^2 \\
& \text{subject to} && \mathbf{E} \in \mathcal{E}, \mathbf{E}(\mathcal{A}) = \mathbf{A} \\
& && \text{rank}(\mathbf{J}\mathbf{E}\mathbf{J}) = 2.
\end{aligned} \tag{6}$$

For this method the penalty function is  $\varphi_2(\mathbf{E}) = \sum_{i,j} (\sqrt{E_{ij}} - d_{ij})^2$ , which more closely matches the terms in the likelihood function (1), and (6) is thus expected to inherit some of the robustness properties of ML estimation. Expanding the objective function in (6) results in

$$\begin{aligned}
& \underset{\mathbf{E}}{\text{minimize}} && \sum_{i,j} (E_{ij} - 2\sqrt{E_{ij}}d_{ij} + d_{ij}^2) \\
& \text{subject to} && \mathbf{E} \in \mathcal{E}, \mathbf{E}(\mathcal{A}) = \mathbf{A} \\
& && \text{rank}(\mathbf{J}\mathbf{E}\mathbf{J}) = 2.
\end{aligned} \tag{7}$$

A relaxed SDP is obtained by introducing an epigraph-like variable  $\mathbf{T}$  and dropping the rank constraint

$$\begin{aligned}
& \underset{\mathbf{E}, \mathbf{T}}{\text{minimize}} && \sum_{i,j} (E_{ij} - 2T_{ij}d_{ij}) \\
& \text{subject to} && T_{ij}^2 \leq E_{ij} \\
& && \mathbf{E} \in \mathcal{E}, \mathbf{E}(\mathcal{A}) = \mathbf{A}.
\end{aligned} \tag{8}$$

From here on, we will call this method EDM with plain ranges (EDM-R).

Remark that the solutions of the initialization techniques described here and in Sections III-A and IV-A are distance matrices. Detailed explanations of how to estimate the spatial coordinates of the sensors and target positions from EDM and the usage of anchors are given in [14]. The basic idea is to use a linear transformation to obtain the Gram matrix  $\mathbf{Y}^T \mathbf{Y}$  from the EDM matrix  $\mathbf{E}$ , from which spatial coordinates  $\mathbf{Y}$  are extracted by singular value decomposition (SVD) up to a unitary matrix. The anchors are then used to estimate the residual unitary matrix by solving a Procrustes problem. As discussed in [14], observation noise can significantly disrupt the estimated sensor/target coordinates through EDM completion and rank truncation, and it was found that much more accurate results are obtained by using those as a starting point for likelihood maximization. MM algorithms are proposed next for iterative likelihood maximization.

### C. SLAT Iterative Refinement: Majorization-Minimization

The key idea of MM is to find, at a certain point  $\mathbf{x}^t$ , a simpler function that has the same function value at  $\mathbf{x}^t$  and anywhere else is larger than or equal to the objective function to be minimized. Such a

function is called a majorization function. By minimizing the majorization function we obtain the next point of the algorithm, which also decreases the cost function [22]. The detailed derivation of MM is given in [14], but it is summarized below for the reader's convenience. Define two convex functions as

$$f_{ij}(\mathbf{x}) = \|x_i - e_j\|, \quad g_{kj}(\mathbf{x}) = \|a_k - e_j\|. \quad (9)$$

Expanding  $f$  and  $g$  in (1) and using first-order conditions on convexity [19],

$$\begin{aligned} \Omega_G(\mathbf{x}) \leq & \sum_{i,j} (f_{ij}^2(\mathbf{x}) - 2d_{ij}(f_{ij}(\mathbf{x}^t) + \langle \nabla f_{ij}(\mathbf{x}^t), (\mathbf{x} - \mathbf{x}^t) \rangle) + d_{ij}^2) \\ & + \sum_{k,j} (g_{kj}^2(\mathbf{x}) - 2d_{kj}(g_{kj}(\mathbf{x}^t) + \langle \nabla g_{kj}(\mathbf{x}^t), (\mathbf{x} - \mathbf{x}^t) \rangle) + d_{kj}^2), \end{aligned} \quad (10)$$

where  $\langle u, v \rangle = u^T v$ , we get the proposed majorization function on the right side of (10), which is quadratic in  $\mathbf{x}$  and easily minimized. Hence the MM iteration

$$\mathbf{x}^{t+1} = \arg \min_{\mathbf{x}} \sum_{i,j} (f_{ij}^2(\mathbf{x}) - 2d_{ij}\langle \nabla f_{ij}(\mathbf{x}^t), \mathbf{x} \rangle) + \sum_{k,j} (g_{kj}^2(\mathbf{x}) - 2d_{kj}\langle \nabla g_{kj}(\mathbf{x}^t), \mathbf{x} \rangle) \quad (11)$$

turns out to be obtained as the solution of a linear system of equations.

#### D. Time-Recursive Position Estimation: SLCP

Suppose that a batch of observations has been processed, and a new target position is to be estimated. We could repeat the previous approach by redefining the new batch as the old one concatenated with the new set of observations. However, this is computationally expensive due to the EDM step. Also, previous estimated positions would be ignored. To alleviate the load we propose a simple methodology to obtain a good initial point which avoids the EDM step. It consists of fixing the previous positions at their estimated values and only estimating the new target position. More precisely, we minimize

$$\Psi_G(y) = \sum_{i=1}^{n+l} (\|b_i - y\| - d_i)^2, \quad (12)$$

where  $y$  is the new target position,  $b_i$  denotes the position of a sensor or anchor, and  $d_i$  is the corresponding range measurement. We propose SLCP [16] to minimize (12), and briefly summarize the method below.

One can readily see that each term in (12) quantifies the distance between the target location and a circle, centered at an anchor or sensor, with radius equal to the measured range, e.g.,  $|\|b_i - y\| - d_i| = \|y - y_i\|$ , where  $y_i$  is located at the intersection of the line connecting  $b_i$  and  $y$  with the circle  $\{z : \|z - b_i\| = d_i\}$ . An equivalent formulation is therefore

$$\begin{aligned} & \text{minimize} \quad \sum_{i=1}^{n+l} \|y - y_i\|^2 \\ & \quad y, y_i \\ & \text{subject to} \quad \|b_i - y_i\| = d_i \quad i = 1, \dots, n+l. \end{aligned} \quad (13)$$



SLCP relies on a complex formulation of (13) for the 2D case, and the problem is then manipulated and relaxed to an SDP of the form

$$\begin{aligned}
& \text{maximize} && t + \frac{1}{n+l} \mathbf{r}^T \mathbf{\Phi} \mathbf{r} \\
& && \mathbf{\Phi}, t \\
& \text{subject to} && \mathbf{\Phi} \succeq 0, \phi_{ii} = 1 \\
& && 4\mathbf{c}^H \mathbf{\Phi} \mathbf{c} \geq t^2,
\end{aligned} \tag{14}$$

where  $\mathbf{r}$  and  $\mathbf{c}$  are constant vectors built from sensor/anchor complex coordinates and measured radii. The solution of the optimization problem (14) is a positive semidefinite matrix, hopefully with near-1 rank. Afterwards, the target coordinates are estimated by SVD of  $\mathbf{\Phi}$  as described in [14].

After an optimal target position is obtained, we return to the cost function (1) or (2) and iteratively refine all the estimates. This incremental or time recursive procedure can be applied to either new targets or sensors.

In a previous paper [14], the source localization method derived in [23], termed Squared Range Least Squares (SR-LS), is proposed as the time recursive method. Note that in [23] *squared* distances are matched, leading to a Trust Region optimization problem. However, as demonstrated in [16], SLCP is a more accurate source localization method and its cost function (12) is better matched to the likelihood function (1). This makes it more convenient for initialization of iterative refinement algorithms, which therefore require fewer iterations to converge and/or are less likely to be trapped in undesirable local extrema.

#### IV. SLAT UNDER LAPLACIAN NOISE

##### A. SLAT Initialization: EDM with Ranges and $\ell_1$ -norm

Among the penalty function approximation methods  $\ell_1$ -norm approximation is known to be robust to outliers [19]. Therefore, the penalty function of the third SLAT initialization method is chosen as  $\varphi_3(\mathbf{E}) = \sum_{i,j} |\sqrt{E_{ij}} - d_{ij}|$ , and the associated optimization problem becomes

$$\begin{aligned}
& \text{minimize} && \sum_{i,j} |\sqrt{E_{ij}} - d_{ij}| \\
& && \mathbf{E} \\
& \text{subject to} && \mathbf{E} \in \mathcal{E}, \mathbf{E}(\mathcal{A}) = \mathbf{A} \\
& && \text{rank}(\mathbf{J}\mathbf{E}\mathbf{J}) = 2.
\end{aligned} \tag{15}$$

This problem is not convex, as the objective function  $|\sqrt{E_{ij}} - d_{ij}|$  is convex when  $\sqrt{E_{ij}} - d_{ij} < 0$ , but concave for  $\sqrt{E_{ij}} - d_{ij} > 0$ . To obtain a convex approximation the objective function is replaced by a

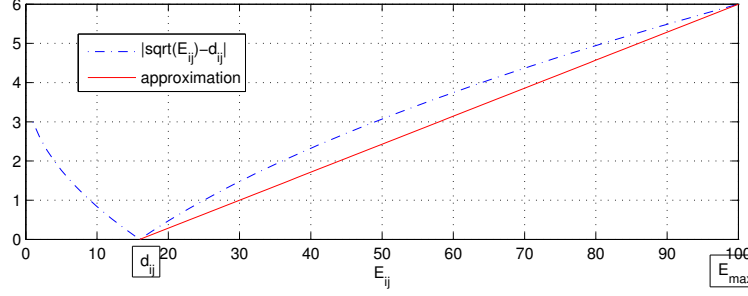


Fig. 1: The value of  $|\sqrt{E_{ij}} - d_{ij}|$  vs  $E_{ij}$ , and the linear approximation of the concave part.

linear approximation

$$a_{ij}E_{ij} + b_{ij}, \quad a_{ij} = \frac{1}{\sqrt{E_{\max}} + d_{ij}}, \quad b_{ij} = -\frac{d_{ij}^2}{\sqrt{E_{\max}} + d_{ij}} \quad (16)$$

in part of the domain where it is concave, as shown in Figure 1. The two functions coincide for  $E_{ij} = d_{ij}^2$  and  $E_{ij} = E_{\max}$ , where the constant  $E_{\max}$  is a practical upper bound on (squared) range measurements. Thus, we replace  $|\sqrt{E_{ij}} - d_{ij}|$  by its convex envelope  $\max\{d_{ij} - \sqrt{E_{ij}}, a_{ij}E_{ij} + b_{ij}\}$  and use the epigraph variable  $\mathbf{T}$  to obtain

$$\begin{aligned} & \text{minimize} \quad \sum_{i,j} T_{ij} \\ & \mathbf{E}, \mathbf{T} \\ & \text{subject to} \quad \max\{d_{ij} - \sqrt{E_{ij}}, a_{ij}E_{ij} + b_{ij}\} \leq T_{ij} \\ & \quad \mathbf{E} \in \mathcal{E}, \mathbf{E}(\mathcal{A}) = \mathbf{A} \\ & \quad \text{rank}(\mathbf{J}\mathbf{E}\mathbf{J}) = 2. \end{aligned} \quad (17)$$

A relaxation of (17) after dropping the rank constraint is

$$\begin{aligned} & \text{minimize} \quad \sum_{i,j} T_{ij} \\ & \mathbf{E}, \mathbf{T} \\ & \text{subject to} \quad (d_{ij} - T_{ij})^2 \leq E_{ij} \\ & \quad a_{ij}E_{ij} + b_{ij} \leq T_{ij} \\ & \quad \mathbf{E} \in \mathcal{E}, \mathbf{E}(\mathcal{A}) = \mathbf{A}. \end{aligned} \quad (18)$$

Note that the first constraint in (18) is not equivalent to  $d_{ij} - \sqrt{E_{ij}} \leq T_{ij}$ , but rather to  $-\sqrt{E_{ij}} \leq d_{ij} - T_{ij} \leq \sqrt{E_{ij}}$ , which amounts to intersecting the original epigraph with the parabolic hypograph  $d_{ij} + \sqrt{E_{ij}} \geq T_{ij}$ . This preserves the convexity of the feasible set and does not change its lower boundary for  $E_{ij} \in [0, E_{\max}]$ , where the optimal point will be found. The constraint can now be readily expressed in

standard form without introducing additional variables, e.g., as an LMI or a second-order cone constraint [24]

$$\begin{bmatrix} 1 & d_{ij} - T_{ij} \\ d_{ij} - T_{ij} & E_{ij} \end{bmatrix} \succeq 0 \quad \text{or} \quad \left\| \begin{bmatrix} 2(d_{ij} - T_{ij}) \\ E_{ij} - 1 \end{bmatrix} \right\| \leq E_{ij} + 1. \quad (19)$$

This technique will be called EDM with ranges and  $\ell_1$ -norm (EDM-R- $\ell_1$ ).

### B. SLAT Iterative Refinement: Weighted Majorization Minimization

Robustness to outliers in the cost function (2) for Laplacian noise is gained at the expense of differentiability. To circumvent that shortcoming we resort to the well-known re-weighted least squares approach [25], which replaces the minimization of (2) with a sequence of minimizations of smooth approximation functions that converge to  $\Omega_L(\mathbf{x})$ . Specifically, (2) is first written as

$$\Omega_L(\mathbf{x}) = \sum_{i,j} u_{ij} (\|x_i - e_j\| - d_{ij})^2 + \sum_{k,j} v_{kj} (\|a_k - e_j\| - d_{kj})^2, \quad (20)$$

with

$$u_{ij} = \frac{1}{\| \|x_i - e_j\| - d_{ij} \|}, \quad v_{kj} = \frac{1}{\| \|a_k - e_j\| - d_{kj} \|}.$$

At time  $t$  the function to be minimized becomes  $\Omega_L^t(\mathbf{x})$ , which has the same form of (20) but the *functions*  $u_{ij}$ ,  $v_{kj}$  above are now replaced by *constants* based on the estimated positions after the previous iteration

$$u_{ij}^t = \frac{1}{\| \|x_i^t - e_j^t\| - d_{ij} \|}, \quad v_{kj}^t = \frac{1}{\| \|a_k^t - e_j^t\| - d_{kj} \|}. \quad (21)$$

An inner optimization loop could now be used to minimize  $\Omega_L^t(\mathbf{x})$  for every  $t$  but, as shown in Appendix A, a single iteration suffices to ensure convergence. With fixed  $u_{ij}^t$ ,  $v_{kj}^t$  the same majorization technique of Section III-C then yields the weighted-MM iteration

$$\mathbf{x}^{t+1} = \arg \min_{\mathbf{x}} \sum_{i,j} u_{ij}^t (f_{ij}^2(\mathbf{x}) - 2d_{ij} \langle \nabla f_{ij}(\mathbf{x}^t), \mathbf{x} \rangle) + \sum_{k,j} v_{kj}^t (g_{kj}^2(\mathbf{x}) - 2d_{kj} \langle \nabla g_{kj}(\mathbf{x}^t), \mathbf{x} \rangle). \quad (22)$$

In practice the weights  $u_{ij}^t$  and  $v_{kj}^t$  must be modified to avoid the possibility of division by zero [26], which in our case is achieved by saturating them at  $10^5$  when computing (22).

### C. Time-Recursive Position Estimation: $SL\ell_1$

The ML source localization problem under Laplacian noise is equivalent to

$$\underset{y}{\text{minimize}} \quad \Psi_L(y) = \sum_{i=1}^{n+l} \| \|y - b_i\| - d_i \| \quad (23)$$

or

$$\begin{aligned} & \underset{y}{\text{minimize}} \quad \left( \sum_{i=1}^{n+l} \left| \|y - b_i\| - d_i \right| \right)^2, \\ & \end{aligned} \tag{24}$$

where  $y$ ,  $b_i$  and  $d_i$  are defined in section III-D. We use ideas from [27] to express the minimization of  $\Psi_L^2$  as a weighted sum of squares.

**Lemma IV-C.1.** *The following problem is equivalent to (24)*

$$\begin{aligned} & \underset{y}{\text{minimize}} \quad \underset{\boldsymbol{\lambda} \in \mathbb{R}^{n+l}}{\text{minimize}} \quad \sum_{i=1}^{n+l} \frac{(\|y - b_i\| - d_i)^2}{\lambda_i} \\ & \text{subject to} \quad \lambda_i > 0, \quad \mathbf{1}^T \boldsymbol{\lambda} = 1. \end{aligned} \tag{25}$$

A proof is given in Appendix B. As claimed in section III-D the difference between the true range and observed range is actually equivalent to the distance between the source position and the point on the circle with center  $b_i$  and radius  $d_i$ . An equivalent formulation is therefore

$$\begin{aligned} & \underset{y, y_i, \boldsymbol{\lambda}}{\text{minimize}} \quad \sum_{i=1}^{n+l} \frac{\|y - y_i\|^2}{\lambda_i} \\ & \text{subject to} \quad \|y_i - b_i\| = d_i \\ & \quad \quad \quad \lambda_i > 0, \quad \mathbf{1}^T \boldsymbol{\lambda} = 1. \end{aligned} \tag{26}$$

If we fix the  $y_i$  and  $\lambda_i$ , the solution of (26) with respect to  $y$  is an unconstrained optimization problem whose solution is readily obtained by invoking the optimality conditions

$$\sum_{i=1}^{n+l} \frac{(y - y_i)}{\lambda_i} = 0 \Rightarrow y^* = \frac{\sum_{i=1}^{n+l} \frac{y_i}{\lambda_i}}{\sum_{i=1}^{n+l} \frac{1}{\lambda_i}}. \tag{27}$$

Geometrically, the first constraint of (26) defines circle equations, which can be compactly described in the complex plane as  $y_i = b_i + d_i e^{j\phi_i}$ . We collect these into a vector  $\mathbf{y} = \mathbf{b} + \mathbf{R}\mathbf{u}$ , where  $\mathbf{b} = [b_1 \ \dots \ b_{n+l}]^T \in \mathbb{C}^{n+l}$ ,  $\mathbf{R} = \text{diag}(d_1, \dots, d_{n+l}) \in \mathbb{R}^{(n+l) \times (n+l)}$ , and  $\mathbf{u} = [e^{j\phi_1} \ \dots \ e^{j\phi_{n+l}}]^T \in \mathbb{C}^{n+l}$ . Using the optimal  $y$ , we get

$$\begin{aligned} & \underset{\boldsymbol{\lambda}, \mathbf{u}}{\text{minimize}} \quad \mathbf{y}^H \boldsymbol{\Pi} \mathbf{y} = (\mathbf{b} + \mathbf{R}\mathbf{u})^H \boldsymbol{\Pi} (\mathbf{b} + \mathbf{R}\mathbf{u}) \\ & \text{subject to} \quad \lambda_i > 0, \quad \mathbf{1}^T \boldsymbol{\lambda} = 1 \\ & \quad \quad \quad |u_i| = 1, \end{aligned} \tag{28}$$

where

$$\begin{aligned}\mathbf{\Pi} &= \begin{bmatrix} \frac{1}{\lambda_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{\lambda_{n+l}} \end{bmatrix} - \frac{1}{\sum_{i=1}^{n+l} \frac{1}{\lambda_i}} \begin{bmatrix} \frac{1}{\lambda_1} \\ \vdots \\ \frac{1}{\lambda_{n+l}} \end{bmatrix} \begin{bmatrix} \frac{1}{\lambda_1} & \cdots & \frac{1}{\lambda_{n+l}} \end{bmatrix} \\ &= \mathbf{\Lambda}^{-1} - \mathbf{\Lambda}^{-1} \mathbf{1} (\mathbf{1}^T \mathbf{\Lambda}^{-1} \mathbf{1})^{-1} \mathbf{1}^T \mathbf{\Lambda}^{-1},\end{aligned}\tag{29}$$

with  $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_{n+l})$ .

Matrix  $\mathbf{\Pi}$  resembles an orthogonal projector. Using the matrix inversion lemma<sup>1</sup> it is seen to be the limiting case  $\mathbf{\Pi} = \lim_{\sigma \rightarrow \infty} (\mathbf{\Lambda} + \sigma \mathbf{1} \mathbf{1}^T)^{-1}$  and thus positive semidefinite. This format is more amenable to analytic manipulations in optimization problems and will be used throughout. The parameter  $\sigma$  is taken as a sufficiently large constant (see Appendix B), although it could also be regarded as an additional optimization variable to ensure adequate approximation accuracy.

We now introduce an epigraph variable  $t \in \mathbb{R}$  in (28), i.e., we minimize over  $t$  and add the constraint  $t - (\mathbf{b} + \mathbf{R}\mathbf{u})^H \mathbf{\Pi} (\mathbf{b} + \mathbf{R}\mathbf{u}) \geq 0$ . Applying Schur complements the constraint may be successively written as

$$\begin{bmatrix} t & (\mathbf{b} + \mathbf{R}\mathbf{u})^H \\ \mathbf{b} + \mathbf{R}\mathbf{u} & \mathbf{\Pi}^{-1} \end{bmatrix} \succeq 0 \quad \Leftrightarrow \quad \mathbf{\Pi}^{-1} - \frac{(\mathbf{b} + \mathbf{R}\mathbf{u})(\mathbf{b} + \mathbf{R}\mathbf{u})^H}{t} \succeq 0. \tag{30}$$

The formulation becomes

$$\begin{aligned}&\text{minimize} && t \\&&& \boldsymbol{\lambda}, \mathbf{u}, t \\&\text{subject to} && \lambda_i > 0, \mathbf{1}^T \boldsymbol{\lambda} = 1 \\&&& |u_i| = 1 \\&&& t\mathbf{\Lambda} + t\sigma \mathbf{1} \mathbf{1}^T \succeq (\mathbf{b} + \mathbf{R}\mathbf{u})(\mathbf{b} + \mathbf{R}\mathbf{u})^H.\end{aligned}\tag{31}$$

Finally, we define  $\mathbf{B} = [\mathbf{b} \ \mathbf{R}]$ ,  $\mathbf{v}^H = [1 \ \mathbf{u}^H]$ ,  $\mathbf{V} = \mathbf{v} \mathbf{v}^H$ , and ignore the rank-1 constraint on the new variable  $\mathbf{V}$  to obtain the relaxed SDP

$$\begin{aligned}&\text{minimize} && t \\&&& \boldsymbol{\beta}, \mathbf{V}, t \\&\text{subject to} && \beta_i > 0, \mathbf{1}^T \boldsymbol{\beta} = t \\&&& V_{ii} = 1, \mathbf{V} \succeq 0 \\&&& \text{diag}(\boldsymbol{\beta}) + t\sigma \mathbf{1} \mathbf{1}^T \succeq \mathbf{B} \mathbf{V} \mathbf{B}^H.\end{aligned}\tag{32}$$

<sup>1</sup> $(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1}$ .

The solution of the optimization problem (32) includes the positive semidefinite matrix  $\mathbf{V}$  from whose first row or column  $\mathbf{u}$  can be extracted directly<sup>2</sup>. to obtain  $\mathbf{y} = \mathbf{b} + \mathbf{R}\mathbf{u}$  and the target coordinates from (27).

As in section III-D, after an optimal target position is obtained we return to the cost function (2) and iteratively refine all the estimates. It is demonstrated in simulation in section V that  $\text{SL}\ell_1$  is more robust to outliers than the SLCP algorithm of section III-D, as its cost function (23) is better matched to the likelihood function (2).

## V. NUMERICAL RESULTS

**Example 1 [Comparison of Batch Initialization Methods]:** To investigate the accuracy of the methods, we set a physical scenario containing four anchors, five unknown sensors, and six target positions in a  $[0, 2] \times [0, 2]$  area. Range measurements are corrupted by additive spatio-temporally white noise with standard deviation  $\sigma_{\text{gaussian}} \in [0.005, 0.03]$ . This noisy observation model may lead to near-zero or negative range measurements, in which case we follow normal practice [6] and set them equal to a small positive constant ( $10^{-5}$  in our simulations). With the chosen noise variances this occurs sufficiently seldom (up to 0.04% of measurements) for its impact on estimation accuracy to be unimportant. Several algorithms are tested (EDM-SR, EDM-R, EDM-R- $\ell_1$ , MM initialized by EDM-SR (EDM-SR+MM), MM initialized by EDM-R (EDM-R+MM) and MM initialized by EDM-R- $\ell_1$  (EDM-R- $\ell_1$ +MM)), and their performances compared according to the total root mean square error (RMSE)

$$\sqrt{\frac{1}{K} \frac{1}{n+m} \sum_{k=1}^K \sum_{i=1}^{n+m} \|x_i - \hat{x}_i^k\|^2}, \quad (33)$$

where  $\hat{x}_i^k$  denotes the  $i$ -th estimated sensor or target position in the  $k$ -th Monte Carlo run for the specific noise realization. In each of  $K = 150$  Monte Carlo runs, a random network is generated according to the physical scenario described above. To assess the fundamental hardness of position estimation, error plots for Gaussian noise also show the total Cramér-Rao Lower Bound (CRLB), calculated as

$$\sqrt{\frac{1}{K} \frac{1}{n+m} \sum_{k=1}^K \text{trace}(\text{CRLB}_k)} \quad (34)$$

for each noise variance, where  $\text{CRLB}_k$  denotes the  $k$ -th diagonal element of the matrix lower bound. The CRLB for anchored and anchor-free localization using ranging information has been studied in

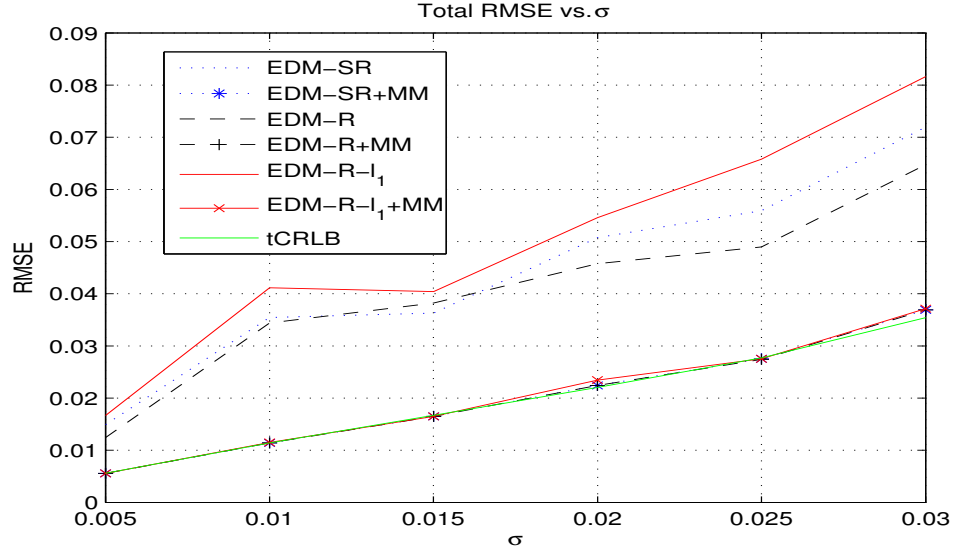
<sup>2</sup>Alternatively,  $\mathbf{u}$  can be obtained by rank 1 factorization of the lower right submatrix of  $\mathbf{V}$  corresponding to  $\mathbf{u}\mathbf{u}^H$ , as in [6], [21].

[28]–[30] for different variance models of range estimation noise. For convenience, the CRLB for our SLAT problem under Gaussian noise is rederived in Appendix C in terms of the notation adopted in this paper. We do not prove unbiasedness of our estimators, a mathematically challenging endeavor that would be required to fully justify benchmarking against the CRLB. In our experimental results, however, we found no clear evidence of bias for small noise levels, where convergence to undesirable extrema of the cost functions is avoided. Figure 2a shows that plain EDM-R has better accuracy than EDM-SR and EDM-R- $l_1$ , although the performance gap closes after iterative refinement by MM. Moreover, MM initialized by the various methods nearly touches the CRLB except when the noise variance is large.

To compare the total RMSE of the algorithms in the presence of outliers, modified range measurements are created according to a “selective Gaussian” model  $d_i = \|\cdot\| + w_i + |\epsilon_i|$ , where  $\epsilon_i$  is a white Gaussian noise term with standard deviation  $\sigma_{\text{outlier}} \in [0.4, 2]$ . The disturbance  $\epsilon_i$  randomly affects only two range measurements, whereas  $w_i$  with  $\sigma_{\text{gaussian}} = 0.01$  is present in all observations. This outlier generating model deviates from the earlier Laplacian assumption, but it is arguably representative of observed range measurements in practical systems [8]. Numerical results under a pure Laplacian model will be presented in Examples 3 and 4. In the presence of high noise and/or outliers, Fig. 2b shows that weighted-MM refinement does not close the performance gap between EDM-R- $l_1$ , EDM-R and EDM-SR initialization because in the latter cases the algorithms converge more often to local minima, thus producing a larger total RMSE.

**Example 2 [Uncertainty Ellipsoids]:** To further examine the accuracy of MM and weighted-MM with different initialization methods, we randomly generated two networks of 10 sensors, 4 anchors and 11 target positions. 100 Monte Carlo runs were used to find the mean and ( $1\sigma$ ) uncertainty ellipsoids of the positions estimated by the methods. The mean and uncertainty ellipsoids for  $\sigma_{\text{gaussian}} = 0.025$  and  $\sigma_{\text{gaussian}} = 0.02/\sigma_{\text{outlier}} = 0.8$  are shown in Figs. 3 and 4, respectively. Again, outliers are randomly added to two range measurements in Fig. 4.

Without outliers (Fig. 3) using EDM-SR, EDM-R, or EDM-R- $l_1$  as an initialization to MM makes the uncertainty ellipsoids shrink dramatically after refinement, yielding very similar means and covariances. These are only displayed in the detail view of Fig. 3b, as they are too small to be shown in Fig. 3a. In the presence of outliers, (Fig. 4) the uncertainty ellipsoids of EDM-SR+wMM are bigger than for other methods and the means of the estimated positions are shifted. Since EDM-R- $l_1$  and EDM-R initializations converge to global extrema most of the time, the means of the positions estimated by weighted MM still approach the true positions and their uncertainty ellipsoids are much smaller than for EDM-SR+wMM. In the presence of outliers this example shows that EDM-R- $l_1$ +wMM is clearly superior to the other



(a) Without outliers

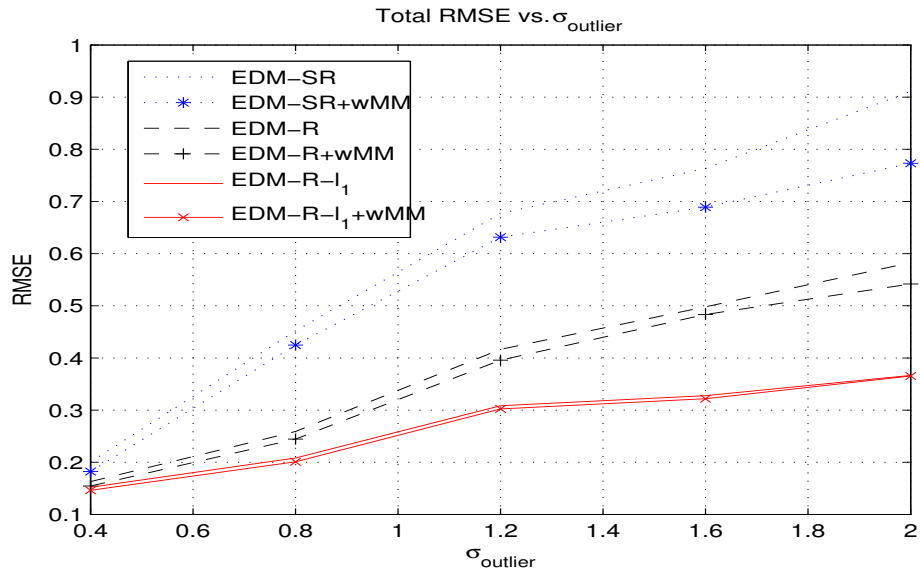
(b) With outliers,  $\sigma_{\text{gaussian}} = 0.01$ 

Fig. 2: Comparison of batch initialization and refinement methods for SLAT.



methods.

*A Note on Practical Computational Complexity of EDM Initialization:* Our experiments were conducted on a machine with an Intel Xeon 2.93 GHz Quad-Core CPU and 8 GB of RAM, using Matlab 7.1, CVX 1.2 and Yalmip 3/SeDuMi 1.1 as a general-purpose SDP solver. CPU times are similar for EDM-SR, EDM-R and EDM-R-11, under 5 seconds for the example described above with  $n = 25$  unknown positions and empirically increasing with  $n^{4.5}$  for larger values of  $n$  ( $< 100$ ). This gives a notion of what network sizes are currently practical for the EDM initialization methods, while keeping in mind that CPU times are known to be unreliable surrogates for intrinsic computational complexity due to dependencies on factors such as machine hardware architecture, operating system, efficiency of numerical libraries, and solver preprocessing. No attempt was made to formulate the EDM completion problems in the most efficient way possible for the SDP solver. For MM type iterative algorithms extremely large problem sizes can be efficiently handled using contemporary numerical algorithms and computing platforms. In our experiments each iteration takes up to about 1 millisecond.

**Example 3 [Comparison of Source Localization Methods for Time-Recursive Initialization]:** In this example  $SL\ell_1$  and SLCP are compared using five anchors. We performed 100 Monte Carlo runs, where in each run the anchor and source positions were randomly generated from a uniform distribution over the square  $[-10, 10] \times [-10, 10]$ . Table I lists the RMSE of source positions under Gaussian noise, with standard deviations  $\sigma_{\text{gaussian}} = 10^{-3}, 10^{-2}, 10^{-1}$ , and 1.  $SL\ell_1$  uses  $\sigma = 10^6$  for the “projector”  $\Pi = (\Lambda + \sigma \mathbf{1}\mathbf{1}^T)^{-1}$ , which is a very conservative value (see Appendix B).

$\sigma_{\text{gaussian}}$	$SL\ell_1$	SLCP
1e-3	1.5e-3	1.3e-3
1e-2	1.41e-2	1e-2
1e-1	0.1720	0.1179
1	1.7922	1.4765

TABLE I: RMSE of  $SL\ell_1$  and SLCP under Gaussian noise.

To compare the algorithms in the presence of outliers, range measurements are created according to  $d_i = \|\cdot\| + v_i$ , where  $v_i$  is a Laplacian noise term with standard deviation  $\sigma_{\text{laplace}} \in [0.2, 1.6]$ . Results are also presented for the alternative selective Gaussian outlier generating model of Examples 1–2 with  $\sigma_{\text{outlier}} \in [0.5, 2]$  and  $\sigma_{\text{gaussian}} = 0.04$ , where outliers only affect measured ranges between the second anchor and the source. Tables IIa and IIb list the RMSE of source positions for these two models. We

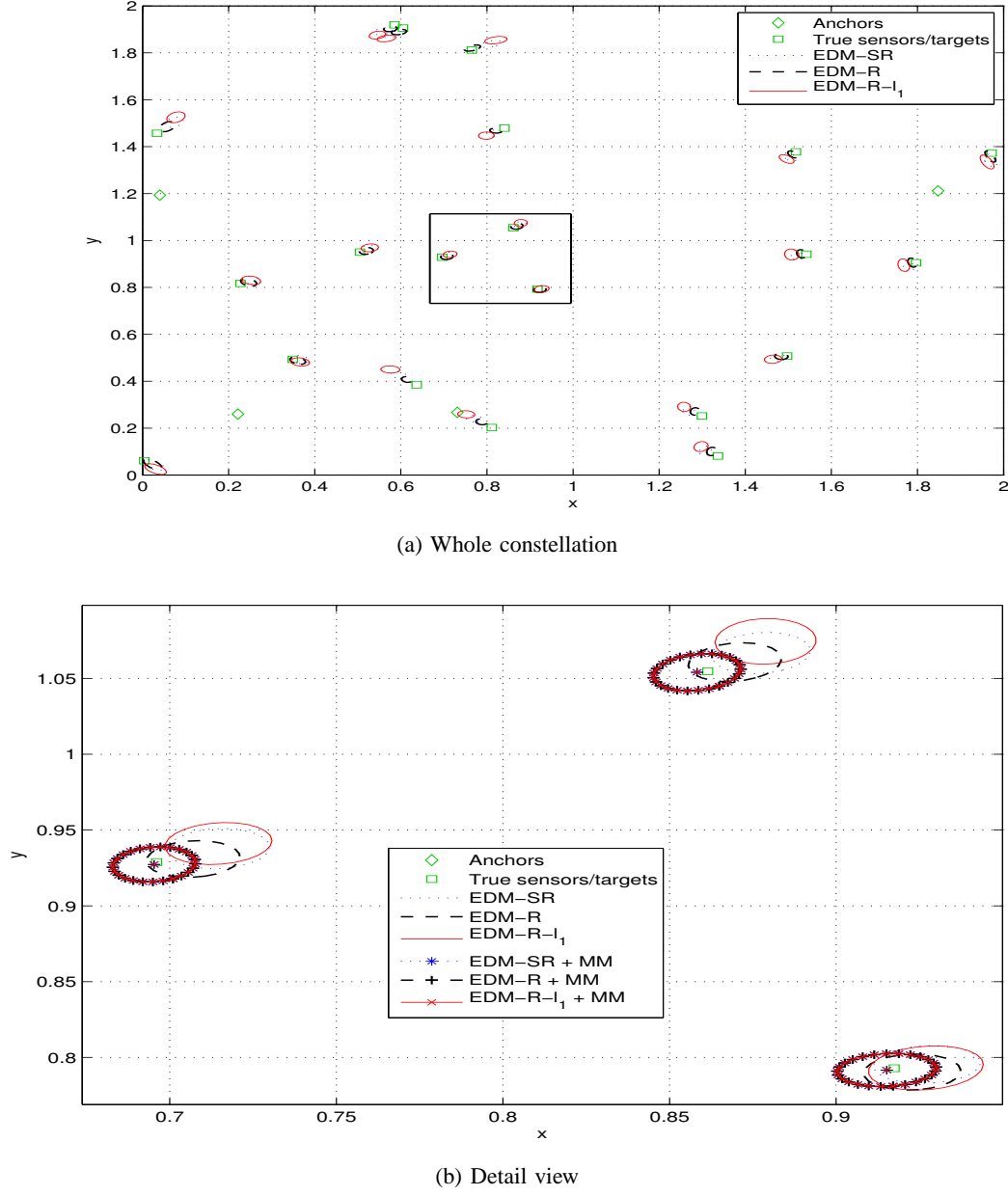
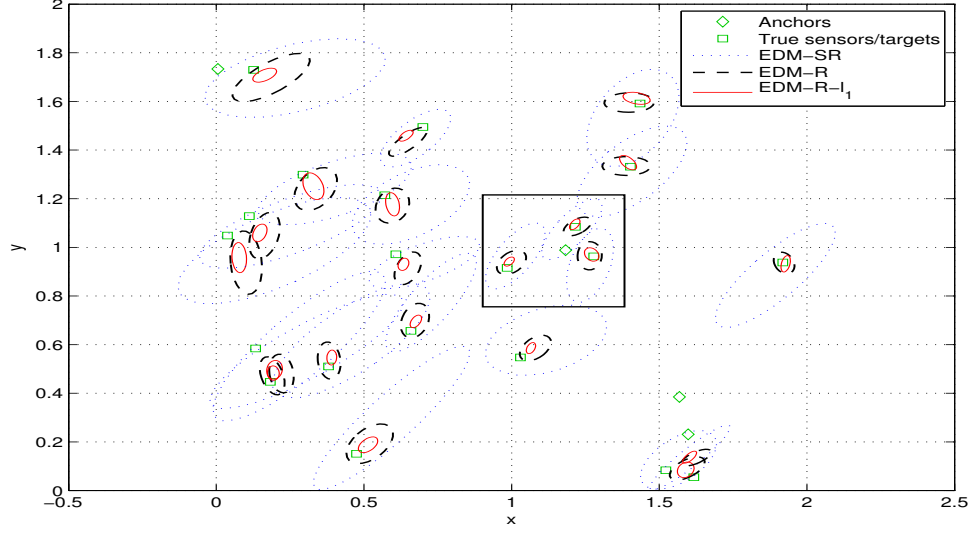
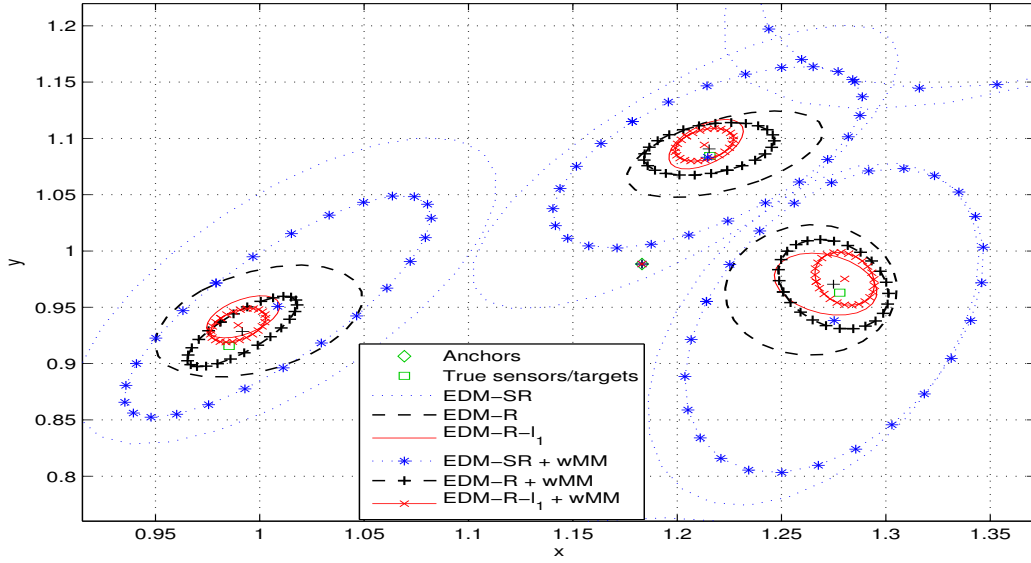


Fig. 3: Mean and uncertainty ellipsoids of MM with different initialization methods and no outliers for  $\sigma_{\text{gaussian}} = 0.025$ .



(a) Whole constellation



(b) Detail view (including an anchor at (1.18; 0.99))

Fig. 4: Mean and uncertainty ellipsoids of weighted MM with different initialization methods and outliers,  $\sigma_{\text{outlier}} = 0.8/\sigma_{\text{gaussian}} = 0.02$ .

$\sigma_{\text{laplace}}$	$\text{SL}\ell_1$	SLCP
0.2	0.2742	0.2757
0.4	0.3990	0.4367
0.8	0.8749	0.9781
1.6	1.5703	1.8872

(a) Laplacian noise

$\sigma_{\text{outlier}}$	$\text{SL}\ell_1$	SLCP
0.5	0.2124	0.2423
1.0	0.3072	0.4734
1.5	0.7067	0.7778
2.0	1.0037	1.4182

(b) Selective Gaussian noise

TABLE II: RMSE of  $\text{SL}\ell_1$  and SLCP in the presence of outliers.

conclude from Tables I and II that the relative accuracy of  $\text{SL}\ell_1$  and SLCP depends on the data generation model. For Gaussian noise the RMSE of  $\text{SL}\ell_1$  is about 30% higher than that of SLCP, whereas in the presence of outliers the situation is reversed, and SLCP exhibits an excess RMSE of 10–30%. Interestingly, the performance gap is actually larger for selective Gaussian outliers, whose generating model does not match the assumptions of  $\text{SL}\ell_1$ . Similarly to Fig. 2b it was found that the differences in initialization accuracy using SLCP or  $\text{SL}\ell_1$  are large enough to prevent closing of the RMSE gap after weighted MM refinement due to convergence to undesirable extrema of the likelihood function.

**Example 4 [Time Recursive Updating]:** This example assesses the performance of the full time-recursive procedure, comprising SLCP or  $\text{SL}\ell_1$  initialization followed by refinement. The network scenario has 16 unknown sensors, 4 anchors and 10 target locations, all randomly positioned. A new target sighting (the 11th one) becomes available and is processed incrementally, i.e., the position is estimated through SLCP or  $\text{SL}\ell_1$  by fixing all the remaining ones, then all estimates are jointly refined. Results are benchmarked against refinement with full batch initialization, which makes a fresh start to the process without using any previous knowledge at every new target position to be estimated, solving different and increasingly large EDM completion problems for ML initialization.

This type of incremental approach was used in [14] with the SR-LS algorithm of [23] and MM refinement for Gaussian noise. SLCP is used here instead of SR-LS because, as shown in [16], it increases the convergence speed of subsequent iterative methods and also alleviates the problem of convergence to local extrema of the ML cost function by providing better initial points than SR-LS does. Figure 5 shows the evolution of the Gaussian cost function  $\Omega_G(\mathbf{x})$  during refinement after ranges to the 11th target position are sensed ( $\sigma_{\text{gaussian}} = 0.04$ ). The time recursive (SLCP)+MM approach takes advantage of previously estimated positions to start with a lower cost than batch (EDM-R)+MM, but reaches the same final error value.

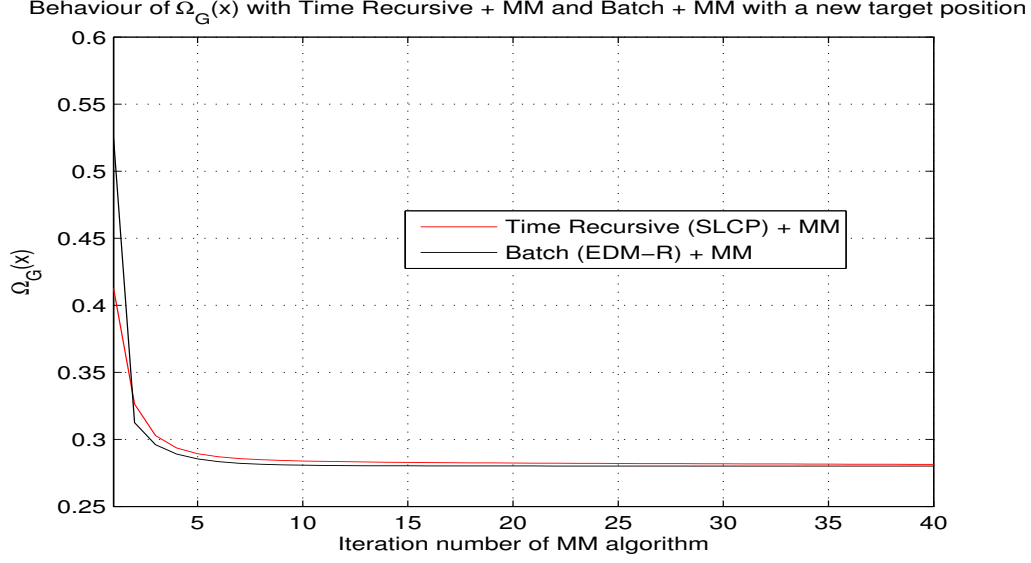


Fig. 5: Evolution of Gaussian cost function  $\Omega_G(\mathbf{x})$  during refinement for EDM-R+MM and SLCP+MM approaches, with  $\sigma_{\text{gaussian}} = 0.04$ .

The same network scenario is adopted in the presence of outliers. Figure 6 shows the evolution of cost function  $\Omega_L(\mathbf{x})$  during refinement for Laplacian outliers ( $\sigma_{\text{laplacian}} = 0.1$ ), whose behavior is similar to the Gaussian case of Fig. 5. In both Gaussian and Laplacian settings refinement yields similar accuracy and convergence speed after batch or time-recursive initializations. Therefore, time-recursive updating is seen to retain the essential features of our EDM-based approach to SLAT, namely, very limited need for *a priori* spatial information and fast convergence, at a fraction of the computational cost.

## VI. CONCLUSION

In this paper, we have presented a ML-based technique to solve a SLAT problem under Gaussian and Laplacian noise. An MM method is proposed to iteratively maximize the non-convex likelihood function, for which a good initial point is required. Therefore, we have investigated two initialization schemes, namely, the batch (EDM) and time-recursive (SLCP/ $\text{SL}\ell_1$ ) approaches that bypass the need for priors on sensor/target locations. After the first block of measurements is obtained, an EDM completion method is used for the first initialization of the sensor network topology. In our experiments this was accomplished reasonably fast (a few seconds) for scenarios with up to about 30 unknown positions. As EDM completion is not scalable, we resort to an alternative, lightweight, incremental initialization scheme as additional target range measurements become available. The SLCP or  $\text{SL}\ell_1$  time recursive methods fix the already

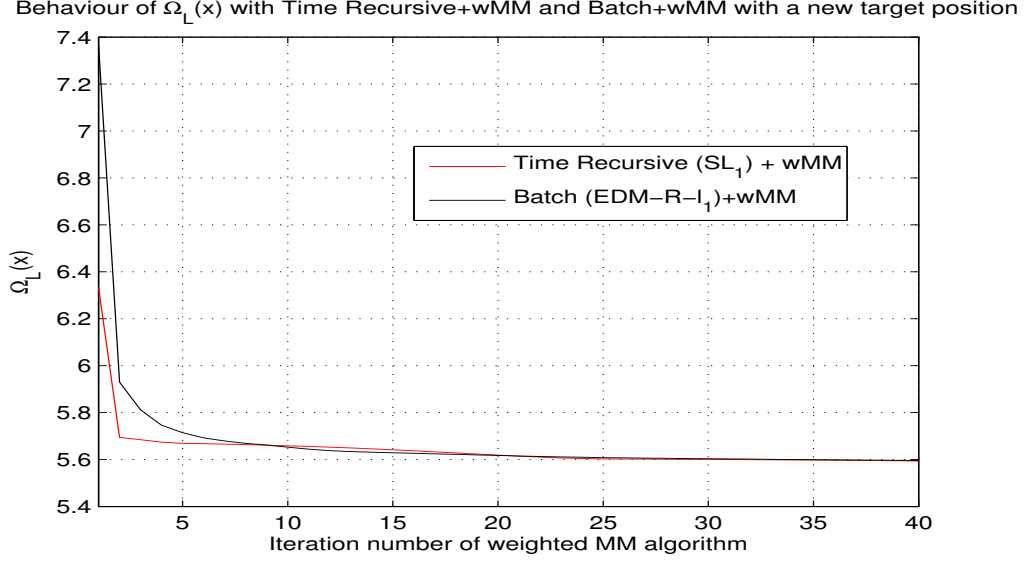


Fig. 6: Evolution of Laplacian cost function  $\Omega_L(\mathbf{x})$  during refinement for EDM-R- $\ell_1$ +wMM and SL $\ell_1$ +wMM approaches, with  $\sigma_{\text{laplacian}} = 0.1$ .

estimated positions whenever a new position is to be determined; afterwards all positions are given as initialization to the likelihood optimization methods.

Simulation results showed that our method nearly attains the Cramér-Rao lower bound under moderate Gaussian noise. In the presence of outliers, EDM-R- $\ell_1$  or SL $\ell_1$  provide more accurate initial position estimates than other existing methods. Moreover, when used as input to iterative refinement methods they provide a good starting point that reduces the probability of convergence to undesirable extrema, yielding improved overall estimation performance. Hence, with this methodology, we obtain a processing structure that is robust to outliers and provides a scalable and accurate solution to the SLAT problem. Importantly, the algorithms based on  $\ell_1$  norm optimization exhibited this robust behavior in simulation not only for Laplacian outliers, but also for an alternative outlier generation technique that did not match the underlying Laplacian modeling assumptions.

## APPENDIX A

## CONVERGENCE OF WEIGHTED MAJORIZATION MINIMIZATION

To prove (local) convergence of the weighted MM iteration (22) the Laplacian cost function (2) is first majorized at time  $t$  by

$$\Gamma_L^t(\mathbf{x}) = \frac{1}{2} \sum_{i,j} \left\{ u_{ij}^t (f_{ij}(\mathbf{x}) - d_{ij})^2 + \frac{1}{u_{ij}^t} \right\} + \frac{1}{2} \sum_{k,j} \left\{ v_{kj}^t (g_{kj}(\mathbf{x}) - d_{kj})^2 + \frac{1}{v_{kj}^t} \right\}, \quad (35)$$

where  $f_{ij}$ ,  $g_{kj}$  and  $u_{ij}^t$ ,  $v_{kj}^t$  are defined in (9) and (21). The inequality  $\Omega_L(\mathbf{x}) \leq \Gamma_L^t(\mathbf{x})$  follows from

$$\begin{aligned} \Gamma_L^t(\mathbf{x}) - \Omega_L(\mathbf{x}) &= \frac{1}{2} \sum_{i,j} \left\{ u_{ij}^t (f_{ij}(\mathbf{x}) - d_{ij})^2 + \frac{1}{u_{ij}^t} - 2|f_{ij}(\mathbf{x}) - d_{ij}| \right\} \\ &\quad + \frac{1}{2} \sum_{k,j} \left\{ v_{kj}^t (g_{kj}(\mathbf{x}) - d_{kj})^2 + \frac{1}{v_{kj}^t} - 2|g_{kj}(\mathbf{x}) - d_{kj}| \right\} \\ &= \frac{1}{2} \sum_{i,j} \left\{ \sqrt{u_{ij}^t} |f_{ij}(\mathbf{x}) - d_{ij}| - \frac{1}{\sqrt{u_{ij}^t}} \right\}^2 + \frac{1}{2} \sum_{k,j} \left\{ \sqrt{v_{kj}^t} |g_{kj}(\mathbf{x}) - d_{kj}| - \frac{1}{\sqrt{v_{kj}^t}} \right\}^2 \geq 0. \end{aligned} \quad (36)$$

It is easy to check that  $\Omega_L(\mathbf{x}^t) = \Gamma_L^t(\mathbf{x}^t)$ , so  $\Gamma_L^t(\mathbf{x})$  has the properties of a true majorization function for the iterate  $\mathbf{x}^t$ . Now the same technique used in (10) is applied to majorize (35) by a convex quadratic function of  $\mathbf{x}$ , yielding

$$\begin{aligned} \Omega_L(\mathbf{x}) &\leq \frac{1}{2} \sum_{i,j} \left\{ u_{ij}^t (f_{ij}^2(\mathbf{x}) - 2d_{ij}f_{ij}(\mathbf{x}^t) - 2d_{ij}\langle \nabla f_{ij}(\mathbf{x}^t), (\mathbf{x} - \mathbf{x}^t) \rangle + d_{ij}^2) + \frac{1}{u_{ij}^t} \right\} \\ &\quad + \frac{1}{2} \sum_{k,j} \left\{ v_{kj}^t (g_{kj}^2(\mathbf{x}) - 2d_{kj}g_{kj}(\mathbf{x}^t) - 2d_{kj}\langle \nabla g_{kj}(\mathbf{x}^t), (\mathbf{x} - \mathbf{x}^t) \rangle + d_{kj}^2) + \frac{1}{v_{kj}^t} \right\}. \end{aligned} \quad (37)$$

As before, equality holds for  $\mathbf{x} = \mathbf{x}^t$ , so the right-hand side of (37) is still a valid majorization function. Discarding constant terms the weighted MM iteration (22) results.

## APPENDIX B

PROPERTIES OF SINGLE-SOURCE LOCALIZATION USING  $\text{SL}\ell_1$ 

*Proof of Lemma IV-C.1:* To streamline the notation we define  $K_i = |||y - b_i|| - d_i|$ , and apply the KKT condition to the inner optimization problem in (25) while fixing  $y$ . The Lagrangian function is

$$L(\boldsymbol{\lambda}, \gamma) = \sum_{i=1}^{n+l} \frac{K_i^2}{\lambda_i} + \gamma(\mathbf{1}^T \boldsymbol{\lambda} - 1). \quad (38)$$

The KKT conditions are

$$\frac{dL}{d\lambda_i} = -\frac{K_i^2}{\lambda_i^2} + \gamma^* = 0, \quad \mathbf{1}^T \boldsymbol{\lambda} = 1. \quad (39)$$

Using (39), we find  $\lambda_i^* = \frac{K_i}{\sum_{i=1}^{n+l} K_i}$  as a solution of the inner optimization problem. Plugging the optimal  $\lambda$  in the cost function of (25) yields  $(\sum_i K_i)^2$ , thus establishing the equivalence with (24). ■

*Approximation accuracy of  $\mathbf{\Pi} = \lim_{\sigma \rightarrow \infty} (\mathbf{\Lambda} + \sigma \mathbf{1}\mathbf{1}^T)^{-1}$ :* To decide how large  $\sigma$  should be, let us first define  $\mathbf{\Pi}(\sigma) = (\mathbf{\Lambda} + \sigma \mathbf{1}\mathbf{1}^T)^{-1}$ . The norm of the difference to the original definition of  $\mathbf{\Pi}$  in (29) is given by

$$\begin{aligned} \|\mathbf{\Pi} - \mathbf{\Pi}(\sigma)\|_F &= \|\mathbf{\Lambda}^{-1} \mathbf{1}[(\mathbf{1}^T \mathbf{\Lambda}^{-1} \mathbf{1})^{-1} - (\mathbf{1}^T \mathbf{\Lambda}^{-1} \mathbf{1} + \sigma^{-1})^{-1}] \mathbf{1}^T \mathbf{\Lambda}^{-1}\|_F \\ &= \frac{\mathbf{1}^T \mathbf{\Lambda}^{-2} \mathbf{1}}{(\mathbf{1}^T \mathbf{\Lambda}^{-1} \mathbf{1})(\sigma \mathbf{1}^T \mathbf{\Lambda}^{-1} \mathbf{1} + 1)}. \end{aligned} \quad (40)$$

Now assume the most unfavorable case with identical  $\lambda_i = \frac{1}{n+l}$ , such that

$$\|\mathbf{\Pi} - \mathbf{\Pi}(\sigma)\|_F = \frac{n+l}{\sigma(n+l)^2 + 1} \leq \epsilon \Rightarrow \sigma \geq \frac{1}{(n+l)\epsilon} - \frac{1}{(n+l)^2}. \quad (41)$$

For  $\epsilon = 10^{-4}$  and  $n+l = 100$ , for example, this yields  $\sigma \geq 10^2 - 10^{-4} \approx 10^2$ , which is quite low and does not raise any numerical issues in commonly available convex optimization solvers. ■

## APPENDIX C

### DERIVATION OF CRLB FOR GAUSSIAN NOISE

The log of the joint conditional pdf for the SLAT problem is (up to an additive constant)

$$\log f(\mathbf{d}|\mathbf{x}) = -\frac{1}{2\sigma^2} \left\{ \sum_{i,j} (\|x_i - e_j\| - d_{ij})^2 + \sum_{k,j} (\|a_k - e_j\| - d_{kj})^2 \right\}, \quad (42)$$

where, similarly to  $\mathbf{x}$ ,  $\mathbf{d}$  denotes the concatenation of all range measurements. Let us define matrices  $\mathbf{M}_{ij}$  and  $\mathbf{N}_j$  that extract individual positions or their differences from the vector of concatenated coordinates  $\mathbf{x}$  as follows

$$\mathbf{M}_{ij}\mathbf{x} = x_i - e_j, \quad \mathbf{N}_j\mathbf{x} = -e_j. \quad (43)$$

Thus, (42) is rewritten as

$$\log f(\mathbf{d}|\mathbf{x}) = -\frac{1}{2\sigma^2} \left\{ \sum_{i,j} (\|\mathbf{M}_{ij}\mathbf{x}\| - d_{ij})^2 + \sum_{k,j} (\|a_k + \mathbf{N}_j\mathbf{x}\| - d_{kj})^2 \right\}. \quad (44)$$

The first derivative of (44) with respect to  $\mathbf{x}$  is

$$\nabla_{\mathbf{x}} \log f(\mathbf{d}|\mathbf{x}) = -\frac{1}{\sigma^2} \left\{ \sum_{i,j} (\|\mathbf{M}_{ij}\mathbf{x}\| - d_{ij}) \frac{\mathbf{M}_{ij}^T \mathbf{M}_{ij}\mathbf{x}}{\|\mathbf{M}_{ij}\mathbf{x}\|} + \sum_{k,j} (\|a_k + \mathbf{N}_j\mathbf{x}\| - d_{kj}) \frac{\mathbf{N}_j^T (a_k + \mathbf{N}_j\mathbf{x})}{\|a_k + \mathbf{N}_j\mathbf{x}\|} \right\}. \quad (45)$$



The second derivative of (44) with respect to  $\mathbf{x}$  is

$$\begin{aligned} \nabla_{\mathbf{x}}^2 \log f(\mathbf{d}|\mathbf{x}) = & -\frac{1}{\sigma^2} \left\{ \sum_{i,j} \left\{ \frac{\mathbf{M}_{ij}^T \mathbf{M}_{ij} \mathbf{x} \mathbf{x}^T \mathbf{M}_{ij}^T \mathbf{M}_{ij}}{\|\mathbf{M}_{ij} \mathbf{x}\|^2} + \frac{\|\mathbf{M}_{ij} \mathbf{x}\| - d_{ij}}{\|\mathbf{M}_{ij} \mathbf{x}\|} \left( \mathbf{M}_{ij}^T \mathbf{M}_{ij} - \frac{\mathbf{M}_{ij}^T \mathbf{M}_{ij} \mathbf{x} \mathbf{x}^T \mathbf{M}_{ij}^T \mathbf{M}_{ij}}{\|\mathbf{M}_{ij} \mathbf{x}\|^2} \right) \right\} \right. \\ & \left. + \sum_{k,j} \left\{ \frac{\mathbf{N}_j^T (a_k + \mathbf{N}_j \mathbf{x}) (a_k + \mathbf{N}_j \mathbf{x})^T \mathbf{N}_j^T}{\|a_k + \mathbf{N}_j \mathbf{x}\|^2} + \frac{\|a_k + \mathbf{N}_j \mathbf{x}\| - d_{kj}}{\|a_k + \mathbf{N}_j \mathbf{x}\|} \left( \mathbf{N}_j^T \mathbf{N}_j - \frac{\mathbf{N}_j^T (a_k + \mathbf{N}_j \mathbf{x}) (a_k + \mathbf{N}_j \mathbf{x})^T \mathbf{N}_j^T}{\|a_k + \mathbf{N}_j \mathbf{x}\|^2} \right) \right\} \right\} \end{aligned} \quad (46)$$

The Fisher information matrix,  $F_{\mathbf{x}}$ , is obtained by taking the negative expected value of the (46) with respect to ranges as [15]

$$F_{\mathbf{x}} = -\mathbb{E}_{\mathbf{d}} \{ \nabla_{\mathbf{x}}^2 \log f(\mathbf{d}|\mathbf{x}) \} = \frac{1}{\sigma^2} \left\{ \sum_{i,j} \frac{\mathbf{M}_{ij}^T \mathbf{M}_{ij} \mathbf{x} \mathbf{x}^T \mathbf{M}_{ij}^T \mathbf{M}_{ij}}{\|\mathbf{M}_{ij} \mathbf{x}\|^2} + \sum_{k,j} \frac{\mathbf{N}_j^T (a_k + \mathbf{N}_j \mathbf{x}) (a_k + \mathbf{N}_j \mathbf{x})^T \mathbf{N}_j^T}{\|a_k + \mathbf{N}_j \mathbf{x}\|^2} \right\}. \quad (47)$$

The CRLB matrix in (34) is taken as the inverse of  $F_{\mathbf{x}}$ .

## REFERENCES

- [1] C. Taylor, H. Shrobe, A. Rahimi, J. Bachrach, and A. Grue, "Simultaneous localization, calibration, and tracking in an ad hoc sensor network," in *Proc. Int. Conf. Inf. Process. Sensor Netw. (IPSN'06)*, Nashville, USA, April 2006, pp. 27–33.
- [2] S. Funiak, C. Guestrin, M. Paskin, and R. Sukthankar, "Distributed localization of networked cameras," in *Proc. Int. Conf. Inf. Process. Sensor Netw. (IPSN'06)*, Nashville, USA, April 2006, pp. 34–42.
- [3] R. Rangarajan, R. Raich, and A. Hero, III, "Euclidean matrix completion problems in tracking and geo-localization," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP'08)*, Las Vegas, Nevada, USA, March 2008, pp. 5324–5327.
- [4] J. Teng, H. Snoussi, and C. Richard, "Decentralized variational filtering for simultaneous sensor localization and target tracking in binary sensor networks," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP'09)*, Taipei, Taiwan, April 2009, pp. 2233–2236.
- [5] R. L. Moses, D. Krishnamurthy, and R. Patterson, "A self-localization method for wireless sensor networks," *EURASIP J. Appl. Signal Process.*, vol. 4, pp. 348–358, 2003.
- [6] P. Biswas, T.-C. Liang, K.-C. Toh, and Y. Ye, "Semidefinite programming approaches for sensor network localization with noisy distance measurements," *IEEE Trans. Autom. Sci. Eng.*, vol. 3, pp. 360–371, 2006.
- [7] W. K. Lui, W. K. Ma, H. C. So, and F. K. W. Chan, "Semi-definite programming approach to sensor network node localization with anchor position uncertainty," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP'09)*, Taipei, Taiwan, April 2009, pp. 2245–2248.
- [8] A. J. Weiss and J. Picard, "Maximum-likelihood position estimation of network nodes using range measurements," *IET Signal Process.*, vol. 2, no. 4, pp. 394–404, 2008.
- [9] S. Srirangarajan, A. H. Tewfik, and Z.-Q. Luo, "Distributed sensor network localization with inaccurate anchor positions and noisy distance information," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP'07)*, vol. 3, Honolulu, Hawaii, USA, April 2007, pp. 521–524.

- [10] M. Vemula, M. F. Bugallo, and P. M. Djurić, "Sensor self-localization with beacon position uncertainty," *Elsevier Signal Process.*, vol. 89, no. 6, pp. 1144–1154, 2009.
- [11] F. K. W. Chan and H. C. So, "Accurate distributed range-based positioning algorithm for wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 57, no. 10, pp. 4100–4105, 2009.
- [12] J. Costa, N. Patwari, and A. Hero, III, "Distributed weighted-multidimensional scaling for node localization in sensor networks," *ACM Trans. Sens. Netw.*, vol. 2, no. 1, pp. 39–64, 2006.
- [13] U. A. Khan, S. Kar, and J. M. F. Moura, "Distributed sensor localization in random environments using minimal number of anchor nodes," *IEEE Trans. Signal Process.*, vol. 57, no. 5, pp. 2000–2016, 2009.
- [14] P. Oğuz-Ekim, J. Gomes, J. Xavier, and P. Oliveira, "ML-based sensor network localization and tracking: Batch and time-recursive approaches," in *Proc. European Signal Process. Conf. (EUSIPCO'09)*, Glasgow, Scotland, August 2009.
- [15] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
- [16] P. Oğuz-Ekim, J. Gomes, J. Xavier, and P. Oliveira, "A convex relaxation for approximate maximum-likelihood 2D source localization from range measurements," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP'10)*, Dallas, Texas, USA, March 2010.
- [17] J. Picard and A. J. Weiss, "Accurate geolocation in the presence of outliers using linear programming," in *Proc. European Signal Process. Conf. (EUSIPCO'09)*, Glasgow, Scotland, August 2009, pp. 2077–2081.
- [18] S. Korkmaz and A. J. van der Veen, "Robust localization in sensor networks with iterative majorization techniques," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP'09)*, Taipei, Taiwan, April 2009, pp. 2049–2052.
- [19] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [20] J. Dattorro, *Convex Optimization and Euclidean Distance Geometry*. Meboo publishers, 2005.
- [21] Y. Ding, N. Krislock, J. Qian, and H. Wolkowicz, "Sensor network localization, Euclidean distance matrix completions, and graph realization," in *Research report CORR 2006-23*, University of Waterloo, Waterloo, ONN2L 3G1, Canada, 2008.
- [22] D. Hunter and K. Lange, "A tutorial on MM algorithms," *The American Statistician*, vol. 58, no. 1, pp. 30–37, 2004.
- [23] A. Beck, P. Stoica, and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1770–1778, May 2008.
- [24] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret, "Applications of second-order cone programming," *Linear Algebra and its Applications*, vol. 284, no. 1-3, pp. 193–228, 1998.
- [25] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Prentice-Hall, 2000.
- [26] P. Rodriguez and B. Wohlberg, "Efficient minimization method for a generalized total variational functional," *IEEE Trans. Image Process.*, vol. 18, no. 2, pp. 322–332, 2009.
- [27] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. SIAM, 2001.
- [28] N. Patwari, A. Hero, III, M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 51, no. 8, pp. 2137–2148, 2003.
- [29] C. Chang and A. Sahai, "Cramer-Rao-type bounds for localization," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 1–13, 2006.
- [30] T. Jia and R. M. Buehrer, "A new Cramer-Rao lower bound for TOA-based localization," in *Proc. Military Commun. Conf. (MILCOM'08)*, November 2008, pp. 1–5.